

MENU

SEARCH

INDEX

DETAIL

JAPANESE

1 / 1

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-060187

(43)Date of publication of application : 06.03.2001

(51)Int.Cl.

G06F 15/16

G06F 12/00

G06F 13/00

G06F 17/30

(21)Application number : 11-236313

(71)Applicant : NIPPON TELEGR & TELEPH CORP  
<NTT>

(22)Date of filing : 24.08.1999

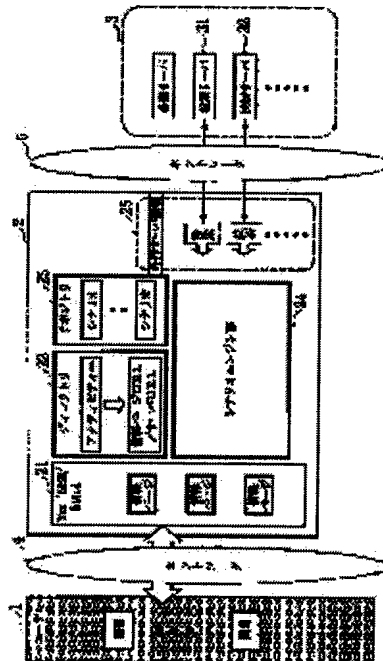
(72)Inventor : MIYAZAKI YU  
NAITO TAKESHI  
HATA KEISUKE  
WATANABE TAKUJI  
NAKAYAMA RYUJI  
AYANO MASAYUKI

## (54) SYSTEM AND METHOD FOR LINKING DISTRIBUTED SERVERS AND RECORDING MEDIUM RECORDING PROGRAM THEREFOR

## (57)Abstract:

PROBLEM TO BE SOLVED: To construct a new service by linking existent operated and disclosed WWW servers while keeping the linkage with a picture watched by a user.

SOLUTION: After data for dispatching information (parameter) to link destination servers 31 and 32 are embedded in a link start application, a scenario engine part 24 and other parts 21-23 are separated and at the time point of a preceding picture, a link application is started. An HTTP method (GET command) is issued to the address of the link server registered in a directory part 22 and the link start application is redirectly started from the engine part 24. In this case, the picture of the link server is displayed. When processing in the link server is advanced and a completion notice parameter is returned, the address of the next link destination server is successively started until receiving the return of the linked final server.



## LEGAL STATUS

[Date of request for examination] 09.08.2002

[Date of sending the examiner's decision of rejection] 02.11.2004

[Kind of final disposal of application other than the examiner's decision of rejection or



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2001-60187

(P2001-60187A)

(43) 公開日 平成13年 3 月 6 日 (2001.3.6)

(51) Int.Cl. <sup>7</sup>	識別記号	F I	テーマコード* (参考)
G 0 6 F 15/16	6 2 0	G 0 6 F 15/16	6 2 0 W 5 B 0 4 5
12/00	5 4 6	12/00	5 4 6 Z 5 B 0 7 5
13/00	3 5 1	13/00	3 5 1 G 5 B 0 8 2
17/30		15/40	3 1 0 C 5 B 0 8 9
			3 3 0 Z

審査請求 未請求 請求項の数 4 O L (全 9 頁)

(21) 出願番号 特願平11-236313

(22) 出願日 平成11年 8 月 24 日 (1999.8.24)

(71) 出願人 000004226

日本電信電話株式会社

東京都千代田区大手町二丁目 3 番 1 号

(72) 発明者 宮崎 祐

東京都千代田区大手町二丁目 3 番 1 号 日

本電信電話株式会社内

(72) 発明者 内藤 岳

東京都千代田区大手町二丁目 3 番 1 号 日

本電信電話株式会社内

(74) 代理人 100077274

弁理士 磯村 雅俊 (外 1 名)

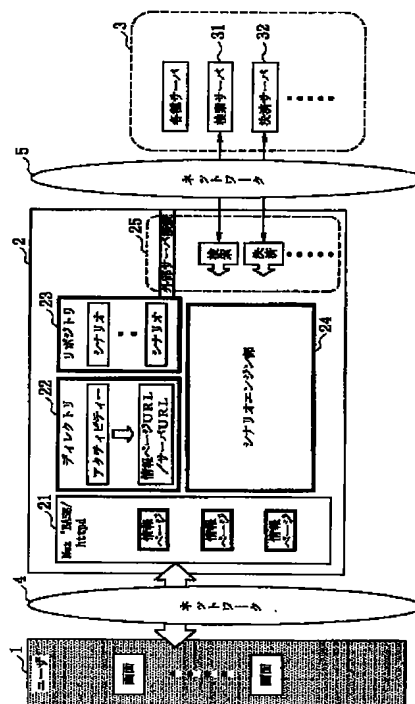
最終頁に続く

(54) 【発明の名称】 分散サーバ連携システムおよび連携方法、ならびにそのプログラムを記録した記録媒体

#### (57) 【要約】

【課題】 ユーザが見ている画面の連携を保ちながら、既存の運用、公開されているWWWサーバと連携させて、新しいサービスを構築する。

【解決手段】 連携開始アプリケーションに、連携先サーバ 3 1, 3 2 に情報 (パラメータ) を渡すデータを埋め込んだ後、シナリオエンジン部 2 4 とその他の部分 2 1 ~ 2 3 を分離し、1 つ前の画面になった時点で、連携アプリケーションに起動をかける。ディレクトリ部 2 2 に登録された連携サーバのアドレスに HTTP メソッド (GET コマンド) を発行し、エンジン部 2 4 から連携開始アプリケーションにリダイレクトで起動をかける。ここで、連携サーバの画面が表示される。連携サーバの処理が進み、完了通知パラメータが返却されたならば、連携する最終のサーバの返却を受けるまで、次の連携先サーバのアドレスに順次起動をかける。



**【特許請求の範囲】**

**【請求項 1】** インターネット上に分散し運用公開されているサーバ上の処理および該処理の画面を連携させ、WWW上で公開されている既存サービスを統合させる分散サーバ連携システムであって、  
任意のサーバ上に、連携する他サーバ間全体を起動・制御するプロセスシーケンスエンジン手段と、該エンジン手段により HTTP メソッドが発行される連携サーバのアドレスを登録するディレクトリ手段と、連携先の各サーバに渡す情報を作成する手段と、その他の処理を行う手段とを配置し、  
該エンジン手段とそれ以外の各手段とを分離した後に、該エンジン手段を起動させることを特徴とする分散サーバ連携システム。

**【請求項 2】** インターネット上に分散し運用公開されているサーバ上の処理および該処理の画面を連携させ、WWW上で公開されている既存サービスを統合させる分散サーバ連携方法であって、  
任意のサーバは、連携先のサーバに渡す情報を作成し、通信プロトコルで上記連携先のサーバに上記情報を渡し、該連携先サーバから終了通知が返却されたならば、同じ処理を繰り返して 1 以上の他サーバの処理および該処理の画面を連携させることを特徴とする分散サーバ連携方法。

**【請求項 3】** 請求項 2 に記載の分散サーバ連携方法において、  
前記連携先サーバに渡す情報として、完了通知 URL、完了通知時パラメータ文字列、および HTTP メソッドを連携開始プログラムに作成し、  
連携先サーバを起動させる 1 つ前の画面になった時点で、上記連携先プログラムにリダイレクトで起動をかけ、  
該連携先サーバのアドレスにコマンドを発行することにより、該連携先サーバに起動をかけ、該連携先サーバの画面を表示させ、  
該連携サーバの処理が進み、該処理が終了した後、完了通知が返却されたならば、連携する最終のサーバの完了通知を受けるまで繰り返し上記処理を行うことを特徴とする分散サーバ連携方法。

**【請求項 4】** 請求項 2 または 3 に記載の分散サーバ連携方法の各処理をプログラム化した後、該プログラムを格納することを特徴としたプログラム記録媒体。

**【発明の詳細な説明】****【0001】**

**【発明の属する技術分野】** 本発明は、各所に分散しているビジネス上の各種処理を WWW サーバ上で連続実行していくためのシステムを構築する環境構築支援技術とそれを利用した統合サービスに関し、特に 1 以上のサーバ上の既存の各アプリケーションおよびその処理画面を順次連携させて、利用者の画面で扱えるようにする分散サ

ーバ連携システムおよび連携方法、ならびにそのプログラムを記録する記録媒体に関する。

**【0002】**

**【従来の技術】** 従来においては、図 2 のシーケンス制御方式 (1) の中の従来方式 (1) に示すように、他サーバとの画面の連携は不可能であった。すなわち、複数のサーバ 41、42、43 は、それぞれサーバ内の各アプリケーションおよび処理画面は容易に連携させることができる (41 内の矢印参照) が、他サーバとの間では連携していないので、それぞれ通信でやりとりすることで制御し合っている。ここでは、サーバ 41 がサーバ 42 のアプリケーション A P を順次呼び出して画面 (丸印) を表示している状態を示している。この場合には、連携していないので、サーバ 42、43 間で通信に時間と手間がかかっている。また、図 3 のシーケンス制御方式 (2) の中の従来方式 (2) に示すように、異なるサーバ全体にわたる画面の連携は不可能であった。すなわち、複数のサーバ 51、52、53 は、他のサーバのワークフロー制御システムを順次連携することができないため、それぞれプロセスを起動させ、データの授受を行い、完了通知を行って互いにやりとりを行っている。

**【0003】** 従来一般に行われている処理アプリケーションおよび処理画面の連携方法、つまり異なる 1 つ以上のサーバにそれぞれ存在する各アプリケーションとその処理画面を順次連続して実行させ、連続画面で出力させるためには、アプリケーション自体にその情報を埋め込むことにより行うことが可能であった。すなわち、アプリケーション A の末尾に、処理が終了したならばアプリケーション B を立ち上げるという命令を埋め込み、さらにアプリケーション B の末尾に処理終了後はアプリケーション C を立ち上げるという命令を埋め込み、順次、連携動作させるアプリケーションの立ち上げ命令を埋め込んでいくことで実現できる。しかし、既存の複数のアプリケーションに手を加えることは困難であり、その結果、アプリケーションの実行手順を変更したり、追加して連携させることは極めて困難であった。また、アプリケーション連携方法において、上述のようにすることでアプリケーションを連携させて動作させることは可能であるが、操作者の画面を連続させることができないため、操作者は複数のアプリケーションの画面をその都度操作する必要があった。すなわち、従来の WWW ブラウザでは、画面にポインタのある画面にしか移動することができなかった。

**【0004】**

**【発明が解決しようとする課題】** 従来より、ビジネスプロセスやワークフロー制御システム等の様々な製品が開発されているが、既に運用、公開されている既存の WWW サーバ上のアプリケーションとその処理画面を順次連携させるための仕組み、およびそのサービスを行っているシステムは未だ見当たらない。そこで、本発明の目的

は、これら従来の課題を解決し、利用者が見ている画面の連携を保ちながら、既存の運用、公開されている異なるWWWサーバ上の複数のアプリケーションおよびその処理画面と連携させる新サービスを構築する分散サーバ連携システムおよび連携方法、ならびにそのプログラムを記録する記録媒体を提供することにある。

#### 【0005】

【課題を解決するための手段】上記目的を達成するため、本発明の分散サーバ連携システムは、(ア) 1つのサーバ上に、連携先のサーバに渡す情報を作成する手段と、通信プロトコルで連携先のサーバに情報を渡す手段とを設けることを特徴としている。また、本発明の分散サーバ連携方法は、(イ) 連携先のサーバに渡す情報を作成し、通信プロトコルで連携先のサーバに情報を渡し、上記動作を繰り返すことで、連携サーバに分散されている画面を統合するサービスを実現する。また、

(ウ) HTTPプロトコルで連携先のサーバに完了通知URL、完了通知時パラメータ文字列、およびHTTPメソッドを連携先サーバに渡す、つまりアプリケーション内にこれらのタグ情報を埋め込んでおくことにより、その情報を連携先サーバに渡し、完了時に通知を行うようにしている。さらに、本発明のプログラム記録媒体は、(エ) 連携先のサーバに渡す情報を作成して、通信プロトコルで連携先サーバに情報を渡す動作を繰り返しながら、連携先アプリケーションを順次起動させていく処理ステップをプログラム化した後、そのプログラムをディスク等の媒体に格納して実現される。

【0006】さらに詳細に説明すると、本発明の分散サーバ連携システムおよび連携方法では、①連携する他サーバ間全体を制御するために、プロセスシーケンス(ビジネスシナリオ)を扱うエンジン手段とプロセスシーケンスの処理を扱う手段とを分離する。②プロセスシーケンスを扱うエンジン手段は、自サーバ内の各処理や他サーバ内のアプリケーションに起動をかける。この起動シーケンスは、各サーバ間をまたがってフロー制御を可能にする。③起動インターフェースはHTTPプロトコルを使用し、その中のリダイレクト機能により画面から各処理や他サーバへ起動がかかるようにすることで、処理画面を連携しながら処理を進めていく。④他サーバへ起動をかけるための情報の受け渡しは、HTTPプロトコルのパラメータに埋め込むことにより行う。このパラメータの受け渡しにより、他の運用しているサーバと連携しようとしている現サーバとの情報調整機能を果すことができる。

#### 【0007】

【発明の実施の形態】以下、本発明の実施例を、図面により詳細に説明する。図1は、本発明の一実施例を示す分散サーバ連携システムの全体構成図である。図1において、1はユーザ端末であり、端末内のアプリケーションによる画面は連続して表示される。3は異なるサーバ

からなるサーバ群を示しており、31は検索サーバ、32は決済サーバであり、その他にも各種サーバが存在する。4、5は通信ネットワークである。2は本発明のシナリオエンジン部24、ディレクトリ22、リポジトリ23、http処理部21および外部サーバに接続するインターフェース部25が設置された1つのサーバである。ディレクトリ22には、情報ページURL(本サーバ2内のアプリケーションのアドレス)とサーバURL(他サーバ内のアプリケーションのアドレス)が登録されており、リポジトリ23には複数のシナリオ(例えば、商品購入シナリオ)が登録されている。http処理部21では、連携先サーバに渡す情報(パラメータ)を作成する。

【0008】連携する他サーバ間全体を制御するために、プロセスシーケンス(例えば、ビジネスシナリオ)を扱うシナリオエンジン部24と、その処理を扱う部分(21, 22, 23)とを分離する。この理由は、連携する順番を制御する部分であるシナリオエンジン部24と他の部分21~23とが分離されていない場合には、シナリオエンジン部24が自分のプログラムと同じように他サーバのプログラムを持ってくるのが不可能になることによる。なお、他サーバへ起動をかけるための情報の受け渡しは、HTTPプロトコルのパラメータに完了通知URLや完了通知時パラメータ文字列やHTTPメソッド(例えば、GETコマンド)をタグとして、本サーバ2内の1つのアプリケーションに埋め込むことで、他サーバへの起動が可能になる。

【0009】図2の中の本方式の図は、本発明の制御方法を説明する図である。本発明においては、本サーバ1から他のサーバ12, 13, 14に対して順番に起動をかけることができる。この場合、HTTPプロトコルのリダイレクト機能を利用する。最初の画面(丸印)からサーバ12のワークフロー管理システムA製品を起動し、これが終了した時点で次の画面(丸印)からサーバ13のワークフロー管理システムB製品を起動し、これが終了した時点で次の画面(丸印)からサーバ14上のアプリケーションAPを起動し、これが終了した時点で最後の画面(丸印)から同じサーバ14上の他のアプリケーションAPを起動する。

【0010】本発明の特徴点は、他サーバへ起動をかけるための情報を本サーバ内のアプリケーションに埋め込み、この情報の受け渡しにより、他の運用しているサーバとの連携を行う点である。図1に示すエンジン部24では、自サーバ2内の各アプリケーションや他サーバに存在するアプリケーションに起動をかける。この起動シーケンスは、各サーバ間をまたがってフロー制御を可能にする。起動インターフェース部25は、HTTPプロトコルを使用し、その中のリダイレクト機能により画面から各アプリケーションや他サーバへ起動をかけることで、処理画面が連携しながら処理が進んでいく。他サー

バへ起動をかけるための情報の受け渡しは、HTTPプロトコルのパラメータにタグ情報を埋め込むことにより行う。このパラメータの受け渡しにより、他の運用しているサーバと連携しようとしている本サーバ2との情報調整機能を果すことができる。

【0011】他サーバへ起動をかけるための情報として、下記の(a)起動用パラメータ、(b)完了通知用パラメータをHTTPプロトコルのパラメータに埋め込む。すなわち、以下のHTTPプロトコルで下記パラメータ((a)起動用パラメータ)を他サーバへ渡し、他サーバのアプリケーションの処理が終了した時点で、下記パラメータ((b)完了通知用パラメータ)により本サーバに返却される。

(a) 起動用パラメータ

(タグ名) (指定項目)

FNUL・・・完了通知URL

FNPR・・・完了通知時パラメータ文字列

FNMT・・・HTTPメソッド

シナリオエンジン部24は、現在実行中のアプリケーションの処理終了後、上記の完了通知URLに指定されたパラメータ、および指定されたHTTPメソッドで、他サーバのアプリケーションにアクセスする。

(b) 完了通知用パラメータ

〈完了通知パラメータ〉&APStat=〈完了ステータス〉&retinfo=〈返却情報〉

上記の式をHTTPプロトコルに従って記述する。シナリオエンジン部24は、他サーバのアプリケーション処理が終了した後、完了通知URLに指定されたパラメータ、HTTPメソッドでその他サーバにアクセスする。

【0012】図4は、本発明の第1の実施例を示す分散サーバ連携方法(決済サーバとの連携)の説明図である。従来より、インターネット上に多数の業務処理サーバが存在しているが、本発明では、そのシステムをそのまま利用し、新しい商品サーバを構築することが可能である。図4は、決済サーバの例を示しており、購入画面により購入ボタンを押すと、上記パラメータとともに決済サーバに起動がかり、決済サーバの処理画面が利用者画面に表示される。決済サーバの処理が終了すると、課金情報とともに元のアプリケーションサーバに処理が戻され、処理終了画面の表示を行う。すなわち、まず画面にブラウザを介してポータルアプリケーション52(商品が多く入っている画面)を表示させ、シナリオエンジン61の商品登録シナリオAPにより利用者に販売受付を行い、次にリダイレクト機能により購入アプリケーション63を起動し、商品販売シナリオにより購入商品を選択して、購入商品をシナリオレジスタ65に順次格納し、次にリダイレクト機能により会員登録・決済アプリケーション64を起動し、支払いの段階に移る。本発明では、上記会員登録・決済アプリケーション64に上記(a)(b)のパラメータを埋め込んでおくこと

により、決済サーバ70の決済サーバアプリケーション71を起動させ、そのアプリケーション71にこのパラメータを渡した後、処理を逐行させ、課金が完了した時点でこのパラメータを本サーバ60の終了通知アプリケーション67に戻すことができる。終了通知アプリケーション67は上記(b)パラメータにより、再度、終了メッセージアプリケーション75を起動するとともに、シナリオエンジン61のアプリケーションの決済ダウンロード終了に戻る。

【0013】図5は、図4の決済サーバの詳細な処理の流れを示す図である。図5において、80は本サーバのシナリオエンジンであって、購入受付、商品購入、課金終了の起動を行い、90は本サーバ(WWWサーバ)の各アプリケーションであり、96~98aはそれぞれパラメータやコマンドであり、100は本サーバのブラウザであり、110が他サーバである決済サーバである。まず、ポータルアプリケーション91を起動して画面に商品を表示し、クリック101によりGETパラメータ96を受け取り、HTTP1.1200 OKのコマンドを送出してOKを表示する。次に、購入アプリケーション62を起動して、GETパラメータ98により購入商品を選択し、次の購入開始アプリケーション93を起動して、埋め込まれた(a)パラメータとGETパラメータ99aにより決済サーバ110にアクセスする。決済サーバのアプリケーションが終了したとき、決済サーバの画面がユーザ側にも表示される。そして、埋め込まれた(a)パラメータの機能により外部起動107が行われ、本サーバの課金格納アプリケーション94に戻される。さらに、(b)パラメータ機能によりリダイレクト109が行われて、課金終了アプリケーション95が起動され、再度、決済サーバ110に送出されることにより、『購入ありがとうございました』の画面を表示させる。

【0014】図6は、本発明の第2の実施例を示す分散サーバ連携方法(検索サーバとの連携)の説明図である。図6は、商品検索サーバの例を示しており、検索開始アプリケーション128から上記(a)(b)パラメータとともに商品検索サーバ131に起動がかり、商品検索サーバ131の検索画面が利用者画面に表示される。この商品検索サーバ131の処理が終了し、利用者の検索作業が終了して、利用者が検索終了ボタンを押すと、商品検索サーバを使用した処理が完了し、本サーバの検索終了アプリケーション129に処理が戻される。ポータルアプリケーション126、購入アプリケーション127等の動作は決済サーバとの連携と同じである。検索としては、例えば、ポータルアプリケーション126の画面によっても、希望の商品が見つからない場合には、商品名や番号を検索キーとして検索を開始する。

【0015】図7は、図6の検索サーバの詳細な処理の流れを示す図である。図7において、140はシナリオ

エンジンであって、販売受付、商品検索、検索終了の起動が行われ、150は本サーバ（WWWサーバ）のアプリケーションであり、157～166はコマンドとパラメータであり、170はブラウザであり、163は商品検索サーバ、180は検索画面である。先ず、ポータルアプリケーション151を起動して画面に商品を表示し、クリック171によりGETパラメータ157を受け取り、HTTP1.1200 OKのコマンドを送出してOKを表示する。次に、受付アプリケーション152を起動して、GETパラメータ159により検索受付

を行い、次の検索開始アプリケーション153を起動して、埋め込まれた（a）パラメータとGETパラメータ161によりフレーム表示177を行って、検索サーバ163にアクセスする。検索サーバ163のアプリケーションが終了したとき、検索サーバ163の画面180がユーザ側にも表示される。そして、埋め込まれた

（a）パラメータの機能によりクリック182が行われ、本サーバの検索格納アプリケーション155に戻される。さらに、（b）パラメータ機能によりリダイレクト184が行われて、検索終了アプリケーション156

が起動され、再度、検索サーバ163に送出されることにより、『検索終了画面』181を表示させる。

【0016】図8は、本発明の一実施例を示す分散サーバ連携方法の処理フローチャートである。先ず、連携する他サーバ間全体を制御するために、本サーバ2上のシナリオエンジン部24とその他の部分21～23を分離する（ステップ101）。次に、各サーバの処理を連結するために、エンジン部24より連携開始アプリケーション（ここでは、商品購入AP）にHTTPプロトコルのリダイレクトで起動をかける（ステップ102）。この時点で、この連携開始アプリケーションは、連携させるサーバの画面の1つ前の画面となっている必要があるため、1つ前の画面であるか否かを判別する（ステップ103）。ここで、連携開始アプリケーションには、HTTPプロトコルで連携先サーバに情報を渡すためのデータが構成されているか否かを確認する（ステップ104）。すなわち、情報（タグ）の指定項目として、完了通知URL、完了通知時パラメータ文字列、およびHTTPメソッド（例えば、GETコマンド）が連携開始アプリケーション内に作成される。未だ構成されていなければ、そのデータを作成してから（ステップ108）、ステップ101に戻って処理を再開する。一方、OKであれば、HTTPプロトコルでディレクトリ部22に登録されている連携サーバのアドレスにGETコマンドを発行する（ステップ105）。GETコマンドにより連携サーバに起動がかかり、連携サーバの画面が表示される。連携サーバの処理が進み、他サーバのアプリケーションが終了後、完了通知URLで示されたURLに指定されたパラメータ、HTTPメソッド（GETコマンド）でアクセスする。そして、連携サーバのアプリケー

ションから完了通知パラメータが返却されたか否かを判別し（ステップ106）、返却された時点で、さらに連携するサーバがあれば、ステップ105に戻ってディレクトリ部22に登録された連携サーバのアドレスにGETコマンドを発行する。このようにして、連携する最終のサーバであるか否かを判別し（ステップ107）、最終のサーバから完了通知パラメータが返却されたとき、処理を終了する。すなわち、連携サーバから本サーバに処理が戻り、その画面も本サーバの処理画面となる。

【0017】図8に示すフローのステップをプログラム化した後、そのプログラムをCD-ROM等の記録媒体に格納しておけば、任意のサーバのコンピュータでこの記録媒体をセットして、プログラムをロードして実行させることにより、本発明を実現することができる。また、ネットワークを介して他のコンピュータから任意のサーバのコンピュータにダウンロードすることによっても本発明を実現することができる。

#### 【0018】

【発明の効果】以上説明したように、本発明によれば、既存のインターネット上のサービスを利用して、既存の運用、公開されているWWWサーバと連携させることができ、さらに新しいサービスを構築することができる、という効果がある。

#### 【図面の簡単な説明】

【図1】本発明の一実施例を示す分散サーバ連携システムの全体構成図である。

【図2】本発明方式および従来方式（1）のシーケンス制御の説明図である。

【図3】従来方式（2）のシーケンス制御の説明図である。

【図4】本発明の第1の実施例を示す分散サーバ連携方法（決済サーバ）のシーケンス制御図である。

【図5】図4の決済サーバとの連携方法の詳細なシーケンス制御図である。

【図6】本発明の第2の実施例を示す分散サーバ連携方法（検索サーバ）のシーケンス制御図である。

【図7】図6の検索サーバとの連携方法の詳細なシーケンス制御図である。

【図8】本発明の一実施例を示す分散サーバ連携方法の処理フローチャートである。

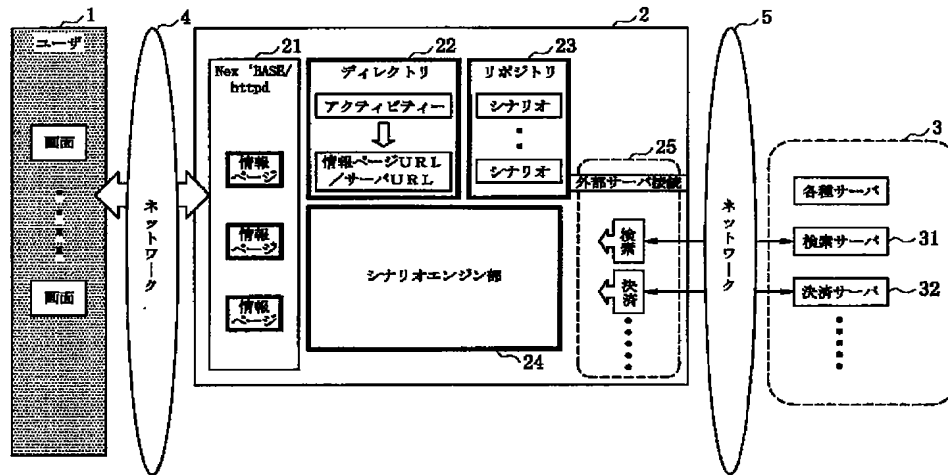
#### 【符号の説明】

1…ユーザ端末、2…連携用サーバ、3…他サーバ群、4, 5…ネットワーク、21…httpd処理部、22…ディレクトリ部、23…リポジトリ部、24…シナリオエンジン部、25…インターフェース部、31…検索サーバ、32…決済サーバ、11, 41…本サーバ制御部、53…ワークフロー管理システムC製品、12, 43, 51…ワークフロー管理システムA製品、13, 52…ワークフロー管理システムB製品、61, 80, 121, 140…シナリオエンジン部、65, 125…シ

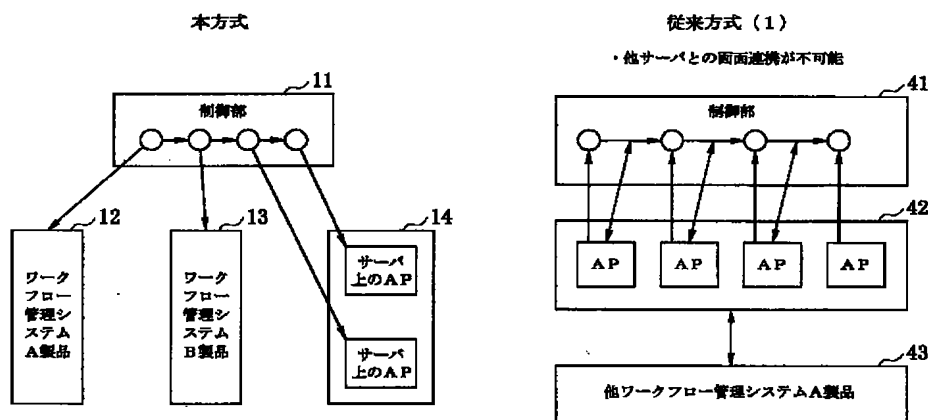
ナリオリポジトリ部、66, 126…ディレクトリ部、62, 91, 126, 151…ポータルアプリケーション、63, 92, 127…購入アプリケーション、64…会員登録、決済アプリケーション、93…購入開始アプリケーション、152…受付アプリケーション、67…終了通知アプリケーション、128, 153…検索開始アプリケーション、70, 110…決済サーバ、10\*

\* 0, 170…ブラウザ、131…情報検索サーバ、163…商品検索サーバ、94…課金格納アプリケーション、95…課金終了アプリケーション、129…検索終了アプリケーション、155…検索格納アプリケーション、156…検索終了アプリケーション、180…検索サーバ画面。

【図 1】



【図 2】

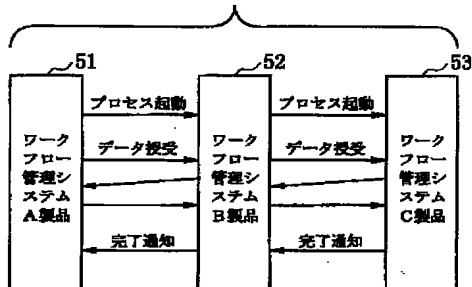




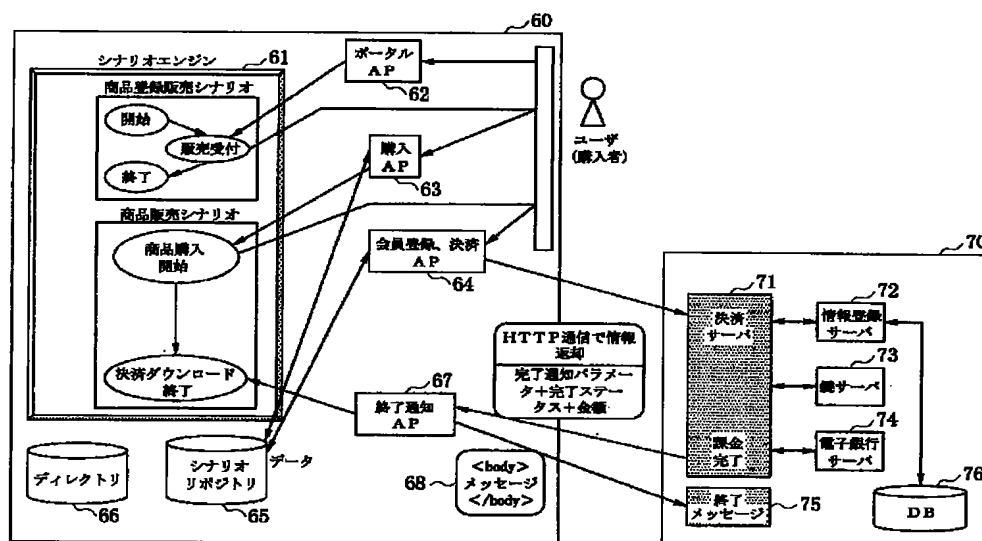
【図3】

## 従来方式(2)

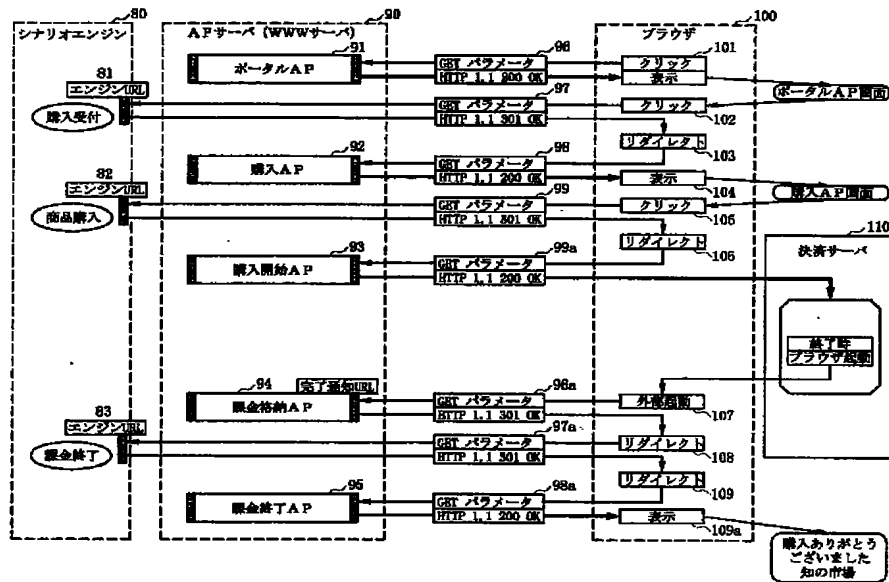
全体にわたる画面の連携が不可能



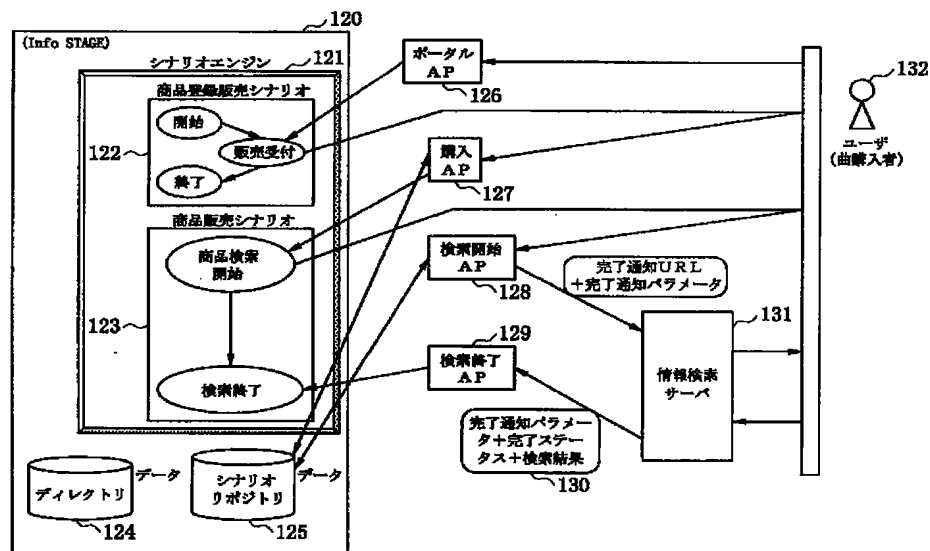
【図4】



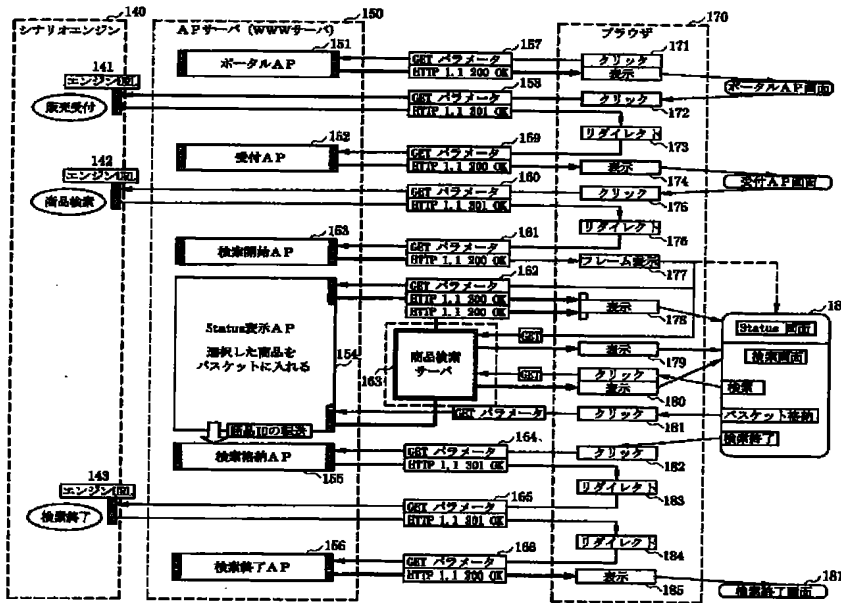
【図5】



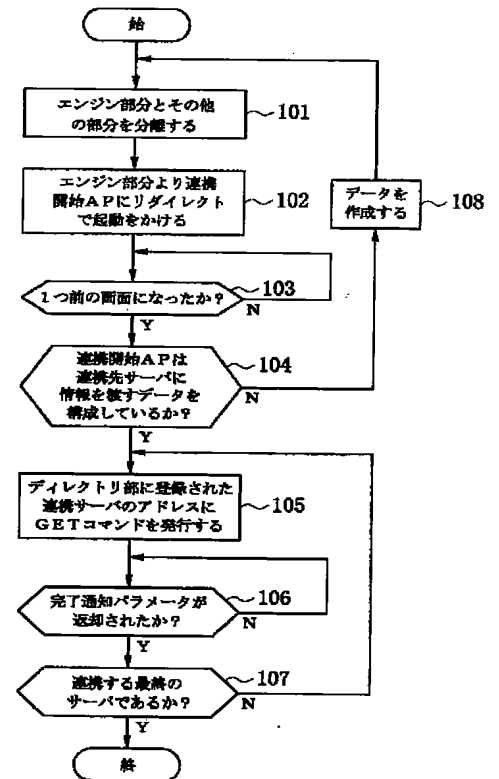
【図6】



【図7】



【図8】



フロントページの続き

(72)発明者 畑 恵介  
東京都千代田区大手町二丁目3番1号 日  
本電信電話株式会社内

(72)発明者 渡邊 拓二  
東京都千代田区大手町二丁目3番1号 日  
本電信電話株式会社内

(72)発明者 中山 隆二  
東京都千代田区大手町二丁目3番1号 日  
本電信電話株式会社内

(72)発明者 綾野 雅之  
東京都千代田区大手町二丁目3番1号 日  
本電信電話株式会社内

Fターム(参考) 5B045 BB03 GG02  
5B075 KK03 PQ42 UU38  
5B082 HA08  
5B089 GA11 GA21 GB04 HA10 JA11  
JB22 KA01 KB07 KB10 LB14